Anisotropic Norm-Oriented Mesh Adaptation for a Poisson problem

G. Brèthes[a], A. Dervieux[a]

[a]*INRIA, Projet Tropics, 2004 route des lucioles - BP 93, 06902 Sophia Antipolis Cedex, France*

Abstract

We present a novel formulation for the mesh adaptation of the approximation of a PDE. The discussion is restricted to a Poisson problem. The proposed formulation extends the goal-oriented formulation, since it is equation-based and uses an adjoint. At the same time, it supersedes it as a solution-convergent method. Indeed, goal-oriented methods rely on the reduction of the error in evaluating a chosen scalar output with the consequence that as mesh size is increased (more degrees of freedom) only this output is proven to tend to its continuous analog, while the solution field itself may not converge. A remarkable throughput of goal-oriented metric-based adaptation is the mathematical formulation of the mesh adaptation problem under the form of the optimization, in the well-identified set of metrics, of a well-defined functional. In the new proposed formulation, we amplify this advantage. We search, in the same well-identified set of metrics, the minimum of a norm of the approximation error. The norm is prescribed by the user and the method allows addressing the case of multi-objective adaptation, like, for example in aerodynamics, adapting the mesh for drag, lift, moment in one shot. In this work we consider the basic linear finite-element approximation and restrict our study to L^2 norm in order to enjoy second-order convergence. Numerical examples for the Poisson problem are computed.

Key words: Compressible flow, goal-oriented mesh adaptation, anisotropic mesh adaptation, adjoint, metric

1. Introduction

This paper addresses anisotropic mesh adaptation. We focus on methods which prescribe a somewhat optimal mesh defined through a parametrization of it by a Riemannian metric. A typical family of optimal metric-based method for CFD is the Interpolation-based/Hessian-based methods. An attractive property of these methods is that they rely on a mathematical optimization principle.

Iso-distribution /equi-repartition Hessian-based methods tend to minimize a Sup or L^∞ norm of the (main term of) interpolation error, with respect to a metric considered in a subset of metrics with a prescribed number of vertices. We refer to the two pioneering works [10, 12]. Multiscale methods tend to minimize the L^p norm ($p < \infty$) of the interpolation error of one or several *sensors* depending on the CFD solution. Cf. [1, 25, 26, 17, 11, 2]. Sensors are field chosen by the user according to their ability to take into account mesh-resolution difficulties of the flow to compute.

The Hessian-based methods appeared as particularly well adapted to the finite-element approximation of elliptic PDEs. It is true, by the projection theorem, that a norm of the approximation error is bounded by the analog norm of the interpolation error, but this concerns the H^1 norm while Hessian-based method concentrate on L^∞ or L^p norms. More generally, while taking into account

the features of the PDE *solution*, these methods do not take into account the features of the *PDE* itself. This is penalizing in the case of systems where sensors need be chosen and weighted by user. However, if sensors are cleverly chosen, a good convergence of the whole approximate solution field to the exact solution field is usually observed.

Taking into account the influence of the PDE on the error through an *equation-based estimate* has been also an important research topic.

An important step for a more justified error evaluation, the formulation of *goal-oriented methods* has first been proposed independently of metrics [4]. It relies on an *a posteriori* estimate. A good synthesis concerning *a posteriori* estimates is [28], see also [13] An interest of *a posteriori estimate* is that it is expressed in terms of the approximate solution, assumed to be available in a mesh adaptation loop. A second interest is that it does not require the use of higher-order (approximate) derivatives, in contrast to truncation analyses. These estimates show accurately where the mesh should be refined. A method for deducing a better anisotropic mesh from an *a posteriori* estimate such as the one from [4] is proposed in [14]. These methods cannot focus on any user-specified error norm, but relies on a particular one, specified by the variational formulation of the EDP. A more popular option is to chose as accuracy target a particular scalar output depending on the PDE solution. Any scalar output can be considered, except that difficulties can arise for the so-called non-admissible ones, according to [3]. An *a posteriori* estimate allows also for building *correctors* for goal-oriented analyses [15]. In [27] the goal-oriented approach is cleverly combined with the correction strategy of [24] and with the Hessian-based metric approach, still minimizing the interpolation error of a user-prescribed sensor.

A priori estimates rely quasi systematically on Taylor series, either through divided differences, or through polynomial approximation of functions. Then approximations of higher-order derivatives of solution need be built from the approximate solution. This is a delicate job since nothing ensures that a higher-order derivative of the approximate solution is a good approximation of the corresponding higher-order derivative of the exact solution. However, since the first recovery methods (for example [30]), many hints in this direction are available.

A remarkable throughput of the goal-oriented metric-based adaptation of [18, 6], is the complete and coherent mathematical formulation of the mesh adaptation problem. Indeed, it takes the form of the optimization with respect to a parameter, the metric, belonging to a well-identified and compact set, of a well-defined functional, namely the error for a prescribed scalar output. This strategy is applied to the discrete case in [29]. In [18, 6], in order to analytically solve the optimum, an a priori analysis is developed. It restricts to the main asymptotic term of the local error in order to exhibit more easily dependance with respect to metric.

Goal-oriented methods have strongly impacted the applications. But due to its formulation, a goal-oriented method shows two limitations. First, it does not naturally extend to several scalar outputs. This "multi-target" issue is well-known and a proposition for addressing it is made in [16]. Second, because they are specialised to a given scalar output, features of the solution field which are not influencing this output may be neglected by the automatic mesh improvement. When a goal-oriented method is used for producing a mesh-converging sequence, the convergence to continuous analog holds for the prescribed scalar output but generally does not hold for the whole flow field itself. To clarify this point, let us consider the mesh adaptive computation of a sonic boom footprint at ground. The functional depends only on pressure at ground. Now, many details of the flow on upper part of the aircraft do not influence the pressure at ground. This is taken into account by the adjoint state which vanishes on these upper regions. Then, in these regions, the adapted mesh is not refined, and the approximation of the flow field does not converge. See an illustration in [21].

In the new norm-oriented formulation proposed in this paper, the user can prescribe a norm or

a semi-norm $|u-u_h|$ of the error, in order to minimise it with respect to the mesh. As a typical example of semi-norm, this can be the sum of square deviations on particular outputs. Let us take again an example in aerodynamics. The semi-norm $|u-u_h| \equiv |C_l(u)-C_l(u_h)|^2+|C_d(u)-C_d(u_h)|^2+|C_m(u)-C_m(u_h)|^2$ will account for minimising the errors on lift, drag, moment measured from flow solution u_h with respect to mesh. The proposed method will ultimately address this kind of semi-norm, assuming that, as for the goal-oriented method, the issue of a non-admisssible norm is solved. As for the goal-oriented method, the proposed method takes into account the PDE features, and in case where a norm is prescribed it produces an approximate solution field which does converge to the exact one in this norm. In this paper, the method is demonstrated with the usual linear finite-element method in 2D. This approximation is first-order accurate for H^1 norm, but second-order accurate for L^2 norm, which we shall consider here. The method relies on the use of a corrector field, and on an *a priori* error estimate from which is extracted the asymptotically largest terms of the local error.

After a formulation of the problem, the derivation of two correctors is proposed in Sec. 2. Next three sections are devoted to the three identified mesh adaptation formulations: Hessian-based, minimizing an interpolation error in Sec. 3, goal-oriented formulation in Sec. 4, and our proposal for a norm-oriented in Sec. 5. Sec. 6 is devoted to a numerical comparison between the two field-convergent formulations, *viz.* Hessian-based and norm-oriented and the paper is completed by a discussion of methods and numerical examples.

2. Two correctors for the Poisson problem

2.1. Notations

Let $V = H_0^1(\Omega)$, Ω being a sufficiently smooth computational domain of $\mathbb{R}^2$. The continuous PDE system is written in short:

$$u \in V \quad Au = f \quad or \quad u \in V \quad \forall\ \phi \in V \quad a(u,\phi) = (f,\phi) \tag{1}$$

where

$$A = -div(\frac{1}{\rho}\nabla) \quad ; \quad a(u,\varphi) = \int \frac{1}{\rho}\nabla u \cdot \nabla\varphi \; dxdy$$

and where $\frac{1}{\rho}$ is a positive, possibly discontinuous, scalar field on Ω. Further, we assume that the bilinear form a is coercive in space V, *i.e.* there exists a positive α such that:

$$a(v,v) \geq \alpha |v|_V^2.$$

Let $\Omega_h = \Omega$ for simplicity, τ_h a triangulation of Ω_h, and V_h be the usual P_1-continuous finite-element approximation space related to τ_h:

$$V_h = \phi_h \in \mathcal{C}^0(\bar{\Omega}) \cap V, \phi_h|_T \text{ is affine } \forall T \in \tau_h.$$

The finite-element discretisation of (1) is written in variational and operational form::

$$u_h \in V_h \quad \text{and} \quad \forall\ \phi_h \in V_h \quad a(u_h,\phi_h) = (f,\phi_h) \tag{2}$$

in such a way that u_h is a linear function of f which we denote:

$$u_h = A_h^{-1} f.$$

We denote by Π_h the usual interpolation operator:

$$\forall v \in \mathcal{C}^0(\bar{\Omega}) \cap V, \quad \Pi_h v \in V_h, \quad \text{and} \quad \forall\ \mathbf{x}_i \text{ vertex of } \Omega_h, \quad \Pi_h v(\mathbf{x}_i) = v(\mathbf{x}_i).$$

Scalar correctors, *i.e.* correctors for scalar outputs $j(u_h)$ depending on the solution, e.g. $j(u_h) = (g,u_h)$, g prescribed have been defined by Giles and co-workers, [15]. Our interest concerns the correction of the unknown field itself.

2.2. A priori corrector for the PDE solution

We observe that:

$$a(u-u_h,\phi_h) = (f-f_h,\phi_h) \quad \forall \phi_h \in V_h.$$

Assuming that the solution u is continuous, we get:

$$a(\Pi_h u - u_h,\phi_h) = a(\Pi_h u - u,\phi_h) + (f-f_h,\phi_h) \quad \forall \phi_h \in V_h. \tag{3}$$

We call $\Pi_h u - u_h$ the *implicit error*. By implicit we mean that it can be obtained through solving a discrete system. It differs from the approximation error by an interpolation error:

$$u - u_h \;=\; \Pi_h u - u_h \;+\; u - \Pi_h u. \tag{4}$$

In order to find an approximate of the implicit error, we need to evaluate the RHS of (3) for any test function ϕ. The second term of RHS is easy to evaluate (we know f and f_h). The first term of RHS can be transformed as follows:

$$\begin{aligned} a(\Pi_h u - u,\phi_h) &= \sum_T \int_T \frac{1}{\rho}\nabla\phi_h \nabla(\Pi_h u - u)\,\mathrm{d}x\mathrm{d}y + (f-f_h,\phi_h) \\ &= \sum_T \int_{\partial T} (\Pi_h u - u)\frac{1}{\rho}\nabla\phi_h\cdot\mathbf{n}\,\mathrm{d}\sigma + (f-f_h,\phi_h). \end{aligned}$$

Then we get:

$$\begin{aligned} a(\Pi_h u - u,\phi_h) &= K(\phi,u_h) \qquad \text{with} \\ K(\phi,u_h) &= \sum_{\partial T_{ij}} \frac{1}{\rho}\nabla(\phi_h|_{T_i} - \phi_h|_{T_j})\cdot\mathbf{n}_{ij}\int_{\partial T_{ij}} (\Pi_h u - u)\,\mathrm{d}\sigma + (f-f_h,\phi_h) \end{aligned} \tag{5}$$

where the last sum is taken for all edges ∂T_{ij} separating triangles T_i and T_j of the triangulation. The unit vector $\mathbf{n}_{ij}$ normal to ∂T_{ij} is pointing outward T_i.

Now we do not know u but u_h. In order to evaluate the interpolation error, we first approximate the Hessian of u by an approximation $H_h(u_h)$ in V_h of the Hessian of u_h. This is done with a Zienkiewicz-Zhu-type ([30]) recovery method defined in the seventh chapter of [23]. Then the evaluation of $\Pi_h u - u$ is built on the edge e_{ij} as a quadratic function vanishing at both extremities of e_{ij}, and of second-derivative in direction e_{ij} defined from $H_h(u_h)$. We replace $\Pi_h u - u$ by $\pi_h u_h - u_h$ where $\pi_h u_h - u_h$ is defined on edge ij as follows:

$$\forall\, \mathbf{x} \in\ e_{ij},\quad (\pi_h u_h - u_h)(\mathbf{x}) = \frac{1}{2}(H_h(u_h)(\mathbf{x}_i) + H_h(u_h)(\mathbf{x}_j))(\mathbf{x}-\mathbf{x}_i)(\mathbf{x}-\mathbf{x}_j)$$

which allow a mid-edge integration on every triangle of Ω_h. We shall see in the sequel that it is useful to apply a similar estimate for the $f - f_h$ term, $f - f_h \approx \pi_h f_h - f_h$.

We define our *a priori implicit corrector* by:

$$\begin{aligned} a(\bar{u}'_{prio},\phi_h) &= K(\phi_h,u_h) \qquad \text{with} \\ K(\phi_h,u_h) \;=\; \sum_{\partial T_{ij}} &\left(\frac{1}{\rho}\nabla\phi_h|_{T_i} - \nabla\phi_h|_{T_j}\right)\cdot\mathbf{n}_{ij} \int_{\partial T_{ij}} (\pi_h u_h - u_h)\,\mathrm{d}\sigma \;-\; (\phi_h,\pi_h f_h - f_h). \end{aligned} \tag{6}$$

Then, we define our *a priori corrector* by:

$$u'_{prio} = \bar{u}'_{prio} - (\pi_h u_h - u_h) \tag{7}$$

built in such a way that:

$$u'_{prio} \approx u - u_h.$$

This corrector is easy to compute but of *a priori* low accuracy.

2.3. Finer-grid defect correction corrector for the PDE solution

When the approximation is far from mesh convergence, probably we have no chance for evaluating the accuracy of a corrector. Let us assume, at the contrary, that the approximation is in its asymptotic mesh convergence phase for the mesh Ω_h under study, of size h. Then this will be also true for a strictly two-times finer embedding mesh $\Omega_{h/2}$, and for our second-order accurate scheme applied to a smooth enough problem, we would have:

$$u_h = A_h^{-1} f_h \quad , \quad u_{h/2} = A_{h/2}^{-1} f_{h/2} \quad \Rightarrow \quad u - u_{h/2} \approx \frac{1}{4}(u - u_h) \tag{8}$$

where u_h and $u_{h/2}$ are respectively the solutions on Ω_h and $\Omega_{h/2}$. We have also:

$$\Pi_h u - \Pi_h u_{h/2} \approx \frac{1}{4}(\Pi_h u - u_h).$$

This motivates the definition of a finer-grid Defect-Correction (DC) corrector as follows:

$$A_h \bar{u}'_{DC} = \frac{4}{3} R_{h/2 \to h}(A_{h/2} P_{h \to h/2} u_h - f_{h/2}) \tag{9}$$

where the residual transfer $R_{h/2 \to h}$ accumulates on coarse grid vertices the values at fine vertices in neighboring coarse elements multiplied with barycentric weights, and $P_{h \to h/2}$ linearly interpolates coarse values on fine mesh. In the case of local singularities, statement (8) is not true for uniform meshes, but we have some hints that it holds almost everywhere for a sequence of adapted meshes, according to [22]. The DC corrector $\bar{u}'_{DC}$ approximates $\Pi_h u - u_h$ instead of $u - u_h$ and can be corrected as the previous one:

$$u'_{DC} = \bar{u}'_{DC} - (\pi_h u_h - u_h). \tag{10}$$

Remark: There exists a second option for this corrector. We start from:

$$a(u_{h/2} - u_h, \phi_h) = (f_{h/2} - f_h, \phi_h).$$

This equation does not define $u_{h/2} - u_h$. In contrast, decomposing $u_{h/2} - u_h = \Pi_{h/2 \to h} u_{h/2} - u_h + u_{h/2} - \Pi_{h/2 \to h} u_{h/2} - u_{h/2}$ where $\Pi_{h/2 \to h} \equiv \Pi_h$ transfers by interpolation onto V_h, then

$$a(\Pi_{h/2 \to h} u_{h/2} - u_h, \phi_h) = a(\Pi_{h/2 \to h} u_{h/2} - u_{h/2}, \phi_h) + (f_{h/2} - f_h, \phi_h).$$

The first term of RHS involves an interpolation error between grids which can be approximated just as in the *a priori* corrector. The above discrete equation can be solved in V_h. Its unique solution $\bar{u}'_{DC2}$ is an approximation of $\Pi_{h/2 \to h} u_{h/2} - u_h$. Then we define:

$$u'_{DC2} = \bar{u}'_{DC2} - (\pi_h u_h - u_h). \square \tag{11}$$

We have not tested this improvement after we remarked that accurate correctors and rather non-accurate ones work with similar efficiency.

3. Interpolation error optimization

3.1. Mesh parametrization

We propose to work in the continuous mesh framework, introduced in [19, 20]. The main idea of this framework is to model discrete meshes by continuous Riemannian metric fields. It allows us to define the adaptation problem as a differentiable optimization problem , *i.e.,* to apply on the class continuous metrics a calculus of variations which cannot be applied on the class of discrete meshes. This framework lies in the class of metric-based methods. A continuous mesh $\mathcal{M}$ of the computational domain Ω is identified to a Riemannian metric field [8] $\mathcal{M} = (\mathcal{M}(\mathbf{x}))_{\mathbf{x}\in\Omega}$ where $\mathcal{M}(\mathbf{x})$ is a symmetric 3×3 matrix. We define the *total number of vertices* of $\mathcal{M}$ as:

$$\mathcal{C}(\mathcal{M}) = \int_\Omega \sqrt{\det(\mathcal{M}(\mathbf{x}))}\, \mathrm{d}\mathbf{x}.$$

Given a continuous mesh $\mathcal{M}$, we shall say that a discrete mesh $\mathcal{H}$ of the same domain Ω is a *unit mesh with respect to* $\mathcal{M}$, if each triangle $K \in \mathcal{H}$, defined by its list of edges $(\mathbf{a}_i\mathbf{b}_i)_{i=1\ldots3}$, verifies:

$$\forall i \in [1,3], \quad \int_0^1 \sqrt{{}^t\mathbf{a}_i\mathbf{b}_i\, \mathcal{M}(\mathbf{a}_i + t\,\mathbf{a}_i\mathbf{b}_i)\, \mathbf{a}_i\mathbf{b}_i}\, \mathrm{d}t \in \left[\frac{1}{\sqrt{2}}, \sqrt{2}\right] .$$

The rest of the paper will try to find the best metric $\mathcal{M}$ from an error analysis which is asymptotic with respect to mesh size.

3.2. Interpolation-based optimal metric

Let u be any smooth enough function defined on Ω. Let $\mathcal{M}$ be a mesh/metric of Ω. We consider only meshes $\mathcal{M}$ involving enough nodes for justifying the replacement of the complete error by its main asymptotic part. The P^1 interpolation error $|\Pi_\mathcal{M} u - u|$ can be approximated in terms of second derivatives of u and of the metric $\mathcal{M}$ by the *continuous interpolation error:*

$$|\Pi_\mathcal{M} u - u| \approx |u - \pi_\mathcal{M} u|$$

with:

$$|u - \pi_\mathcal{M} u|(\mathbf{x}) \;=\; \frac{1}{10}\mathrm{trace}(\mathcal{M}^{-\frac{1}{2}}(\mathbf{x})\, |H_u(\mathbf{x})|\, \mathcal{M}^{-\frac{1}{2}}(\mathbf{x})) \tag{12}$$

where $|H_u|$ is deduced from H_u by taking the absolute values of its eigenvalues. Starting from:

$$\|u - \pi_\mathcal{M} u\|_{\mathbf{L}^p(\Omega_h)} = \left(\int_\Omega \Big(\mathrm{trace}\big(\mathcal{M}^{-\frac{1}{2}}(\mathbf{x})|H_u(\mathbf{x})|\mathcal{M}^{-\frac{1}{2}}(\mathbf{x})\big)\Big)^p \mathrm{d}\mathbf{x}\right)^{\frac{1}{p}} \tag{13}$$

we define as optimal metric the one which minimizes the right hand side under the constraint of a total number of vertices equal to a parameter N. After solving analytically this optimization problem, we get the unique optimal $(\mathcal{M}_{\mathbf{L}^p}(\mathbf{x}))_{\mathbf{x}\in\Omega}$ as:

$$\mathcal{M}_{\mathbf{L}^p} = \mathcal{K}_p(H_u) \;\text{ with }\; \mathcal{K}_p(H_u) = D_{\mathbf{L}^p}\, (\det|H_u|)^{\frac{-1}{2p+2}}\, |H_u| \;\text{ and }\; D_{\mathbf{L}^p} = N^{\frac{2}{2}} \left(\int_\Omega (\det|H_u|)^{\frac{p}{2p+2}}\right)^{-\frac{2}{2}}, \tag{14}$$

where $D_{\mathbf{L}^p}$ is a global normalization term set to obtain a continuous mesh with complexity N and $(\det|H_u|)^{\frac{-1}{2p+2}}$ is a local normalization term accounting for the sensitivity of the $\mathbf{L}^p$ norm. In the

case of an adaptation loop for solving a Partial Differential Equation, a continuous function u is not available, but an approximate solution $u_{\mathcal{M}}$. In that case, the continuous interpolation error (12) is replaced by:

$$|u_{\mathcal{M}} - \pi_{\mathcal{M}} u_{\mathcal{M}}|(\mathbf{x}) \;=\; \frac{1}{10}\text{trace}(\mathcal{M}^{-\frac{1}{2}}(\mathbf{x})\,|H_{u_{\mathcal{M}}}(\mathbf{x})|\,\mathcal{M}^{-\frac{1}{2}}(\mathbf{x})) \tag{15}$$

where $H_{u_{\mathcal{M}}}$ is an approximate Hessian evaluated with the recovery method.

According to the continuous mesh framework, statement (14) defines directly a continuous optimal metric. In practice, solving (14) is done by approximation, *i.e.* in a discrete context with a couple (mesh, solution) denoted $(\mathcal{H}_{\mathcal{M}}, u_{\mathcal{M}})$ and iteratively through the following fixed point:
Step 1: compute the discrete state $u_{\mathcal{M}}$ on mesh $\mathcal{H}_{\mathcal{M}}$,
Step 2: compute sensor $s_{\mathcal{M}} = s(u_{\mathcal{M}})$ and optimal metric $\mathcal{M}^{opt}_{inter} = \mathcal{K}_p(H_{\mathcal{M}}(s_{\mathcal{M}}))$,
Step 3: put $\mathcal{M} = \mathcal{M}^{opt}_{inter}$, generate a new mesh $\mathcal{H}_{\mathcal{M}} = \mathcal{H}_{\mathcal{M}^{opt}_{inter}}$ and go to 1, until convergence.

In the above algorithm, the continuous Hessian of s is replaced by an approximate Hessian $H_{\mathcal{M}}(s_{\mathcal{M}})$, evaluated by the patch-recovery approximation defined in [23]. In our Hessian-based numerical examples, the L^2 case, $p = 2$, has been considered. The above notation $\mathcal{K}_p$ will also be used in the next sections for $p = 1$.

4. Implicit a priori error estimate

In contrast to a corrector as defined in Section 2, an asymptotic upper bound of the approximation error should allow an easier error reduction by minimisation of its norm with respect to the metric. In our PDE discretisation notations, we henceforward replace in the discretization index h by the index $\mathcal{M}$ which holds for any unit mesh of the metric $\mathcal{M}$. The implicit *a priori* estimate (5) then writes:

$$\begin{aligned} &a(\pi_{\mathcal{M}} u_{\mathcal{M}} - u_{\mathcal{M}}, \phi) = K(\phi, u_{\mathcal{M}}) \quad \forall\ \phi \in\ V_{\mathcal{M}} \qquad \text{with} \\ &K(\mathcal{M}, \phi, u) \;=\; \sum_{\partial T_{ij}} \frac{1}{\rho}(\nabla\phi|_{T_i} - \nabla\phi|_{T_j}) \cdot \mathbf{n}_{ij} \int_{\partial T_{ij}} (\pi_{\mathcal{M}} u - u)\ \mathrm{d}\sigma \;-\; (\phi, \pi_{\mathcal{M}} f - f) \end{aligned} \tag{16}$$

where T_{ij} are the triangles of a unit mesh for $\mathcal{M}$, and the proposed corrector is expressed with the discrete solution:

$$\begin{aligned} a(\bar{u}'_{prio}, \phi) &= K(\phi, u_{\mathcal{M}}) \quad \forall\ \phi \in\ V_{\mathcal{M}} \qquad \text{with} \\ K(\mathcal{M}, \phi, u_{\mathcal{M}}) &= \sum_{\partial T_{ij}} \frac{1}{\rho}(\nabla\phi|_{T_i} - \nabla\phi|_{T_j}) \cdot \mathbf{n}_{ij} \int_{\partial T_{ij}} (\pi_{\mathcal{M}} u_{\mathcal{M}} - u_{\mathcal{M}})\ \mathrm{d}\sigma \;-\; (\phi, \pi_{\mathcal{M}} f - f), \\ u'_{prio} &= \bar{u}'_{prio} - (\pi_{\mathcal{M}} u_{\mathcal{M}} - u_{\mathcal{M}}). \end{aligned} \tag{17}$$

We now restart from (5). The following result is proven in [5]:

Lemma 4.1. *We assume that the metric anisotropy is bounded by a positive number. For any smooth couple of functions (u, ϕ), where u is not necessarily a solution of (1), we have the following bound:*

$$\left|\int_\Omega \frac{1}{\rho}\nabla(u - \Pi_{\mathcal{M}} u)\nabla\Pi_{\mathcal{M}}\varphi d\mathbf{x}\right| \;\preceq\; K \int_\Omega \left(\frac{1}{\rho}\bar{\rho}(H(\varphi))\ |u - \Pi_{\mathcal{M}} u| \;+\; |\phi|\ |\Pi_{\mathcal{M}} f - f|\right) d\mathbf{x} \tag{18}$$

where $A \preceq B$ holds for a majoration asymptotically valid, i.e. $A \leq B + o(A)$. Expression $\bar{\rho}(H(\varphi))$ holds for the largest (in absolute value) eigenvalue of the Hessian $H(\varphi)$ of φ. □

The next section shows two ways in using this estimate.

5. Equation-based adaptation

5.1. Scalar output "goal-oriented" analysis

The goal-oriented analysis relies on the minimization of the error $\delta j_{goal}(\mathcal{M})$ committed on a scalar output $j = (g, u)$, error which we simplify as follows:

$$\delta j_{goal}(\mathcal{M}) = |(g, u - u_{\mathcal{M}})| = |(g, \Pi_{\mathcal{M}} u - u_{\mathcal{M}} + u - \Pi_{\mathcal{M}} u)|. \tag{19}$$

The term $u - \Pi_{\mathcal{M}} u$, similar to the main term of the Hessian-based adaptation in Section 3.2, can be explicitly approached in the same way. The term $\Pi_{\mathcal{M}} u - u_{\mathcal{M}}$ will be transformed via an equation, *i.e.* and adjoint state. Let us define the discrete adjoint state $u^*_{g,\mathcal{M}}$:

$$\forall \psi \in \ V_{\mathcal{M}}, \quad a(\psi_{\mathcal{M}}, u^*_{g,\mathcal{M}}) = (\psi_{\mathcal{M}}, g). \tag{20}$$

Then:

$$\delta j_{goal}(\mathcal{M}) \approx |a(u^*_{g,\mathcal{M}}, \Pi_{\mathcal{M}} u - u_{\mathcal{M}}) + (g, u - \Pi_{\mathcal{M}} u)|$$

And introducing the continuous interpolation error (15):

$$\delta j_{goal}(\mathcal{M}) \preceq |a(u^*_{g,\mathcal{M}}, \Pi_{\mathcal{M}} u - u_{\mathcal{M}})| + |g||\pi_{\mathcal{M}} u_{\mathcal{M}} - u_{\mathcal{M}}|$$

Now we integrate by parts, and apply Lemma 4.1, introducing the discrete extension of the interpolation error. We get

$$\delta j_{goal}(\mathcal{M}) \ \preceq \ \int_\Omega \left(\ \big[\frac{1}{\rho}\bar{\rho}(H(u^*_{g,\mathcal{M}})) + |g|\big] \ |\pi_{\mathcal{M}} u_{\mathcal{M}} - u_{\mathcal{M}}| \ + \ |u^*_{g,\mathcal{M}}| \ |\pi_{\mathcal{M}} f - f| \ \right) \ \mathrm{d}\Omega.$$

It is then reasonable to try to minimize the RHS of this inequality instead of the LHS. But this involves still some difficulty due to the dependancy of adjoint state $u^*_{g,\mathcal{M}}$ with respect to $\mathcal{M}$. We shall further simplify our functional by freezing, during a part of the algorithm, the adjoint state. The idea is that, when we change the parameter $\mathcal{M}$, $u^*_{g,\mathcal{M}}$ is close to its (non-zero) continuous limit and is not much affected, in contrast to the interpolation errors $|\pi_{\mathcal{M}} u_{\mathcal{M}} - u_{\mathcal{M}}|$ and $|\pi_{\mathcal{M}} f - f|$. We then consider, for a given $\mathcal{M}_0$, the following optimum problem:

$$\min_{\mathcal{M}} \int_\Omega \left(\ \big[\frac{1}{\rho}\bar{\rho}(H(u^*_{g,\mathcal{M}_0})) + |g|\big] \ |\pi_{\mathcal{M}} u_{\mathcal{M}} - u_{\mathcal{M}}| \ + \ |u^*_{g,\mathcal{M}_0}| \ |\pi_{\mathcal{M}} f - f| \ \right) \ \mathrm{d}\Omega.$$

This will produce an optimum:

$$\mathcal{M}_{opt,\mathcal{M}_0} = \arg\min_{\mathcal{M}} |tr(\mathcal{M}^{-1/2} \left(\big[\frac{1}{\rho}\bar{\rho} H(u^*_{g,\mathcal{M}_0}) + |g|\big] |H_u| + |u^*_{g,\mathcal{M}_0}||H_f| \right) \mathcal{M}^{-1/2})|.$$

Observing that in the integrand

$$H_{goal,0} = [\frac{1}{\rho}\bar{\rho}(H(u^*_{g,\mathcal{M}_0})) + |g|] \ |H_u| \ + \ |u^*_{g,\mathcal{M}_0}| \ |H_f|$$

is a positive symmetric matrix, we can apply the above calculus of variation and get:

$$\mathcal{M}_{opt,\mathcal{M}_0} = \mathcal{K}_1(\ [\frac{1}{\rho}\bar{\rho}(H(u^*_{g,\mathcal{M}_0})) + |g|] \ |H_u| \ + \ |u^*_{g,\mathcal{M}_0}| \ |H_f|).$$

This solution can then be introduced in a fixed-point loop and will produce the solution of:

$$\mathcal{M}_{opt,goal} = \mathcal{K}_1(\ [\frac{1}{\rho}\bar{\rho}(H(u^*_{g,\mathcal{M}_{opt,goal}})) + |g|] \ |H_u| \ + \ |u^*_{g,\mathcal{M}_{opt,goal}}| \ |H_f|).$$

Let us precise how the discrete algorithm is organised:
Step 1: compute the discrete state $u_{\mathcal{M}}$ on mesh $\mathcal{H}_{\mathcal{M}}$,
Step 2: compute the discrete adjoint state $W^*_{\mathcal{M}}$,
Step 3: compute optimal metric $\mathcal{M}^{opt}_{goal}(W_{\mathcal{M}})$,
Step 4: put $\mathcal{M} = \mathcal{M}^{opt}_{goal}(W_{\mathcal{M}})$, generate a new mesh $\mathcal{H}_{\mathcal{M}} = \mathcal{H}_{\mathcal{M}^{opt}_{goal}(W_{\mathcal{M}})}$ and go to 1, until convergence.

The adaptation of this process to the Euler model of Gas Dynamics is studied in [21] for the steady case and in [7] for the unsteady case.

5.2. Norm-based functional

We are now interested by the minimization of the following expression with respect to the mesh $\mathcal{M}$:

$$\delta j(\mathcal{M}) = ||u - u_{\mathcal{M}}||^2_{L^2(\Omega)}. \tag{21}$$

We observe that:

$$\delta j(\mathcal{M}) = (g_{\mathcal{M}}, u - u_{\mathcal{M}}). \tag{22}$$

Let us define the discrete adjoint state $u^*_{\mathcal{M}}$:

$$\forall \psi \in \ V_{\mathcal{M}}, \quad a(\psi_{\mathcal{M}}, u^*_{\mathcal{M}}) = (\psi_{\mathcal{M}}, g_{\mathcal{M}}). \tag{23}$$

Then, similarly to Section 5.1, we have to solve the following optimum problem.

$$\min_{\mathcal{M}} \int_\Omega \left(\ [\frac{1}{\rho}\bar{\rho}(H(u^*_{\mathcal{M}})) + |g|] \ |\pi_{\mathcal{M}} u_{\mathcal{M}} - u_{\mathcal{M}}| \ + \ |u^*_{\mathcal{M}}| \ |\pi_{\mathcal{M}} f - f| \ \right) \ \mathrm{d}\Omega.$$

Exactly as for Section 5.1, we freeze the dependancy of the adjoint state.

$$\min_{\mathcal{M}} \int_\Omega \left(\ [\frac{1}{\rho}\bar{\rho}(H(u^*_{\mathcal{M}_0})) + |g|] \ |\pi_{\mathcal{M}} u_{\mathcal{M}} - u_{\mathcal{M}}| \ + \ |u^*_{\mathcal{M}_0}| \ |\pi_{\mathcal{M}} f - f| \ \right) \ \mathrm{d}\Omega.$$

$$\mathcal{M}_{opt,\mathcal{M}_0} = \mathcal{K}_1(\ [\frac{1}{\rho}\bar{\rho}(H(u^*_{\mathcal{M}_0})) + |g|] \ |H_u| \ + \ |u^*_{\mathcal{M}_0}| \ |H_f|).$$

In order to get the final norm-oriented optimum $\mathcal{M}_{opt,norm}$ we shall:

Step 1: first solve the linearised corrector system:

$$a(\bar{u}'_{prio,\mathcal{M}}, \phi) = \sum_{\partial T_{ij}} \ (\nabla\phi|_{T_i} - \nabla\phi|_{T_j}) \cdot \mathbf{n}_{ij} \ \int_{\partial T_{ij}} (\pi_{\mathcal{M}} u_{\mathcal{M}} - u_{\mathcal{M}}) \ \mathrm{d}\sigma \ - \ (\phi, \pi_{\mathcal{M}} f_{\mathcal{M}} - f_{\mathcal{M}}). \tag{24}$$

where mesh is unit for $\mathcal{M}$, and $\pi_{\mathcal{M}} u_{\mathcal{M}} - u_{\mathcal{M}}$ is expressed in terms of metric and Hessian, as in (6).

Put $u'_{prio,\mathcal{M}} = \bar{u}'_{prio,\mathcal{M}} - (\pi_{\mathcal{M}} u_{\mathcal{M}} - u_{\mathcal{M}})$.

Step 2: then solve the adjoint system:

$$a(\psi, u^*_{prio,\mathcal{M}}) = (u'_{prio,\mathcal{M}}, \psi) \tag{25}$$

Step 3: finally put:

$$\mathcal{M}^{(\alpha+1)} = \mathcal{K}_1([|u'_{prio}| + \frac{1}{\rho}\bar{\rho}H(u^*_{prio})]\ |H_{u_{\mathcal{M}}}| + |u^*_{prio}||H_f|) \tag{26}$$

the three-step process being re-iterated until we get a fixed point $\mathcal{M}_{opt,norm} = \mathcal{M}^{(\infty)}$. Note that the fields $\bar{u}'_{prio,\mathcal{M}}$ and $u^*_{prio,\mathcal{M}}$ are correction fields and can be re-used on a finer mesh after multiplication by the inverse ratio of the number of nodes.

6. Numerical examples

We restrict our study to a benchmark of two-dimensional Poisson problems. We conjecture that the two following mesh adaptation methods produce L^2 convergent solutions to continuous. The first method, the Hessian-based method (with $p = 2$), is just heuristically relying on usual finite-element estimates. The second method, our novel norm-oriented method, is directly built on the minimisation of the L^2 error norm. We do not consider goal-oriented applications, for which examples of computations can be found in [21] and [7]. As already remarked, the convergence of goal-oriented solutions to continuous is definitively questionable.

6.1. Numerical features

In [9], a mesh-adaptative full-multigrid (FMG) algorithm relying on the Hessian-based adaptation criterion is designed. We first describe in short this algorithm for the Hessian-based option. A sequence of numbers N_k of vertices is specified, from a coarse mesh to finer one $N_0 = N, N_1 = 4N, N_2 = 16N, N_3 = 64N, ...$ For each mesh size N_k, a sequence of adapted meshes of size N_k is build by iterating the following loop:
(1) computing a solution,
(2) computing the optimal metric,
(3) building the adapted mesh,
In (1), a multi-grid V-cycle is applied to a sufficient convergence. In (2), approximations of the Hessians are performed as in [21]. When changing of mesh, an interpolation is applied in order to enjoy a good initial condition. About 4 adaptations iterations are applied at each mesh fineness N_k.

The extension of the above loop to norm-oriented adaptation consists of replacing the single Hessian evaluation by:
- the computation of the corrector, using MG and the best available (interpolated to current mesh) previous evaluation,
- the computation of the adjoint, using MG and also the best available (interpolated to current mesh) previous evaluation,
- the evaluation of (26).

Let us discuss computer efficiency. In the demonstrator of [9], a particular feature is the stopping criterion of FMG, which applies to the convergence of the solution of the unique system solved, *i.e.* the system under study, $u = A^{-1}f$. It is then possible to enjoy a better and better initial condition and control the iterative and approximation errors convergence. Consequently, it was possible in [9] to show that mesh adaptation carries large improvement not only in terms of accuracy for a given number of vertices, but also in terms of accuracy for a given computational time.

In contrast, the method proposed in this paper involves three systems to solve: (1) the system under study, $u = A^{-1}f$, (2) the corrector system, (3) the adjoint system. Now the correctors for different mesh sizes are not approximations of a unique continuous field (and same for the adjoints which have the correctors as RHS). For obtaining a computational efficiency similar to [9], we need to go deeper in the choice and analysis of the corrector. The idea would be manage in such a way that $N\ u'$ is converging to a continuous field which does not depend on the number of vertices N. In our opinion a study like this goes a little further than the object of the present paper. The numerical study proposed now restricts to mesh convergence, *i.e.* error as functions of number of vertices.

6.2. 2D Boundary layer

This test case is taken from [14]. We solve the Poisson problem $-\Delta u = f$ dans $]0,1[\times]0,1[$ with Dirichlet boundary conditions and a right-hand side f chosen for having:

$$u(x,y) = [1 - e^{-\alpha x} - (1 - e^{-\alpha})x]4y(1-y).$$

The coefficient α is chosen equal to 100. The graph of the solution is depicted in Figure 1. Before applying our mesh adaptive algorithm, it is interesting to evaluate the accuracy of our correctors. We choose a 161×161 uniform mesh and show in Figure 2 and Figure 3 the cut of $u - u_h$ compared with the cut of u'. We observe that the *a priori* corrector does its job in a correct but inaccurate manner while the DC one is rather accurate. We have also observed that the DC corrector does not consume notably more CPU than the *a priori* one. Therefore we keep this option for the rest of the test case. In Figure 4, we show a set of FMG calculations for the considered test case. The

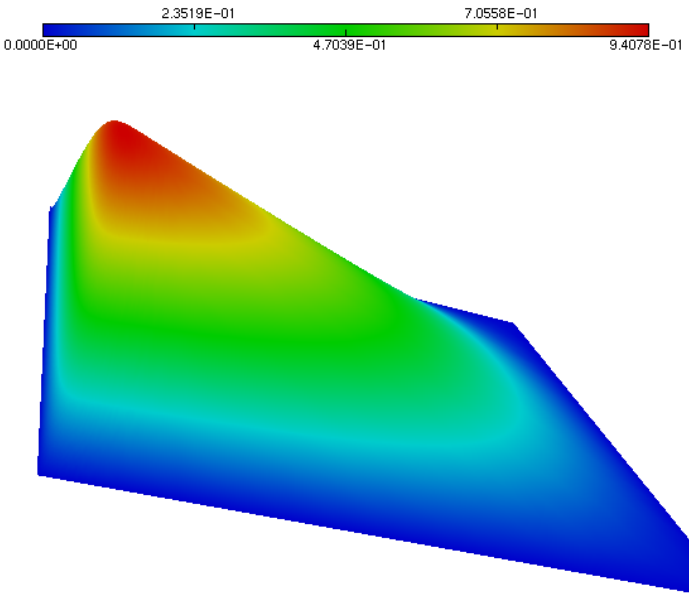

Figure 1: Fully 2D Boundary layer test case : sketch of the solution.

number of vertices of the successive meshes are supported by the horizontal axis, from 120 vertices to 30,000 vertices. The vertical axis gives the L^2-norm of the approximation error $|u - u_h|_{L^2}$ obtained

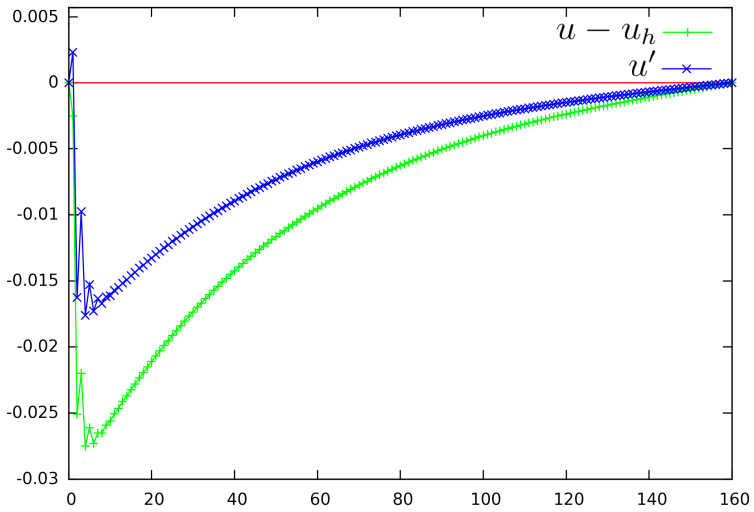

Figure 2: Fully 2D Boundary layer test case : comparison of error cuts for $y = 0.5$: plus signs (+) depict the approximation error $u - u_h$ and crosses ($\times$) depict the *a priori* corrector u'_{prio}. The corrector is able to correct about 60% of the approximation error.

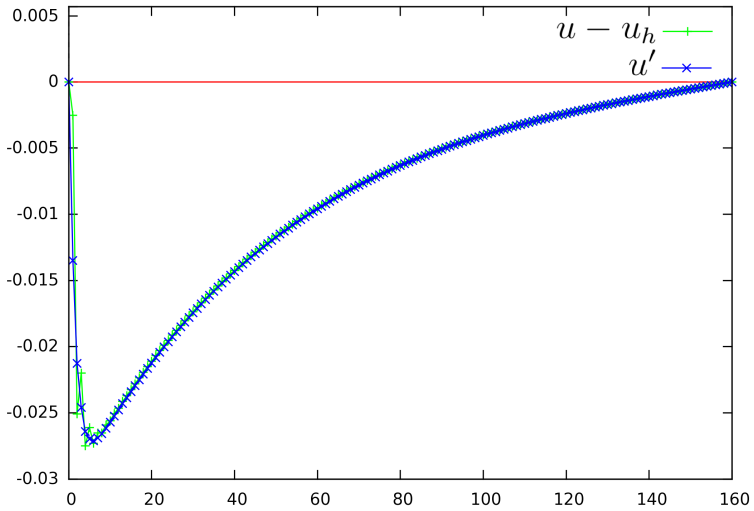

Figure 3: Fully 2D Boundary layer test case : comparison of error cuts for $y = 0.5$: plus signs (+) depict the approximation error $u - u_h$ and crosses ($\times$) depict the Defect-Correction corrector u'_{DC}. The corrector is able to correct about 95% of the approximation error.

on the mesh. Its variation with respect to number of vertices is compared in Figure 4 for the three following algorithms: (a) the uniform-mesh FMG, and (b) the Hessian-based adaptative FMG, and (c) the norm-oriented adaptive FMG. We observe that both adaptation methods carry an important improvement with respect to uniform-grid FMG (25921 vertices on finest mesh). For essentially the same number of vertices (32318), the Hessian option gives an error divided by 47. The norm-oriented option appears as better, with an error divided by 208 with 29485 vertices.

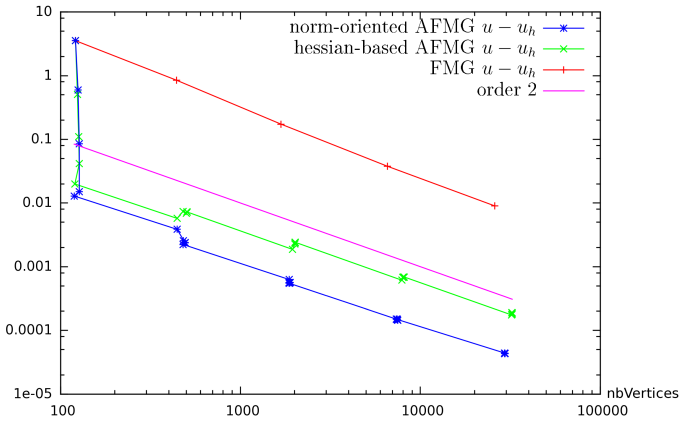

Figure 4: Fully 2D Boundary layer test case: convergence of the error norm $|u - u_h|_{L^2}$ as a function of number of vertices in the mesh for (+) non-adaptive FMG, (×) Hessian-based adaptive FMG, (∗) norm-oriented adaptive FMG.

We also propose an *a posteriori* measure of the correctors efficiency by comparing the convergence of our norm-oriented adaptation equipped with either corrector with the same method in which the corrector is replaced by $u - u_h$. Of course that latter algorithm is not a mesh adaptation method since we assume that we already know the exact solution. In Figure 5, we observe that the error convergence for the three computations are very close to each other. This confirms the interest of the two proposed correctors.

Lastly, when using the proposed method in usual conditions, *i.e.* without knowing the analytic solution, it can be interesting to have an evaluation of the final error $u - u_h$ by using the corrector which has been introduced. Figure 6 shows a comparison of the virtual error computed from the corrector with the actual error. We observe that the prediction is useful, giving the good order of magnitude. For the finer mesh of $30,000$ vertices, the predicted error is $2.14\ 10^{-5}$ while the actual error is $4.31\ 10^{-5}$. For most meshes, the prediction remains somewhat optimistic (smaller than the actual error).

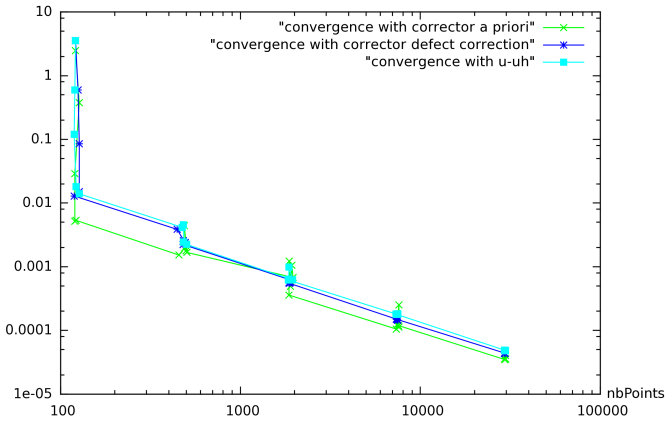

Figure 5: Fully 2D Boundary layer test case: efficiency of the norm-oriented adaptive FMG based on ($\times$) the *a priori* corrector or on the ($*$) Defect-Correction one, compared with ($\square$) a virtual adaptation controlled by $u - u_h$.

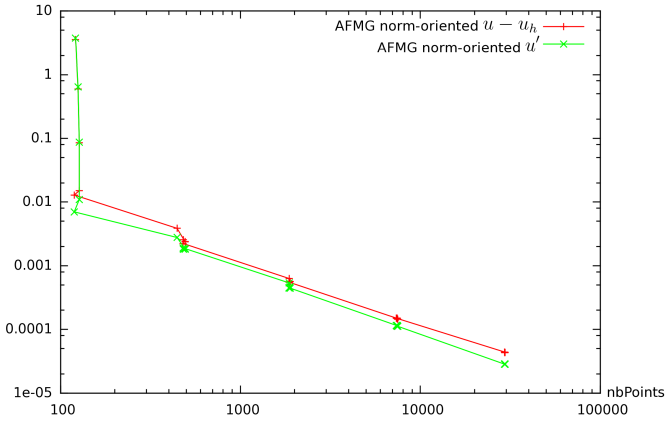

Figure 6: Fully 2D Boundary layer test case: comparison of the predicted error norm proposed by the method, curve with ($\square$), with the exact error norm ($\times$).

6.3. Bubble-like test case with thick interface

We are interested by a Poisson problem the solution of which is a function u equal to 1 on a disk and to 0 in the rest of the domain. This function is the prototype of the pressure in a multi-fluid flow involving capillary forces. The source term is a Dirac derivative. We smoothen this computation by defining a thickness ε of an annular region separing the two subdomains (outside the disk, inside the disk) and in which u is smoothly varying from 0 to 1 as shown in figure 7. If (x, y) is located inside the annular region, $u(x, y)$ is given by the formula: $u(x, y) = \frac{1}{2} + \frac{1}{2}sin(\frac{\pi\psi}{\varepsilon})$ with $\psi = 0.25 - \sqrt{(x_C - x)^2 + (y_C - y)^2}$. From this solution, a right-hand side f is computed. Given a mesh, vertex values of f_h are interlopated from the analytic f. As a result, for rather coarse meshes, the zone where f is not zero can be simply missed and f_h can be zero even in the neighborhood of the high values of f. We consider first a quite large thickness of $\varepsilon = 0.1$. An approximate solution u_h is shown in Figure 8. As for the previous test case, we first evaluate the accuracy of the corrector. We choose a uniform mesh 161×161 and show in Figure 9 and Figure 10 the cut of $u - u_h$ compared with the cut after correction, that is $u - u_h - u'_h$. We observe that both *a priori* and Defect-Correction correctors do an accurate job.

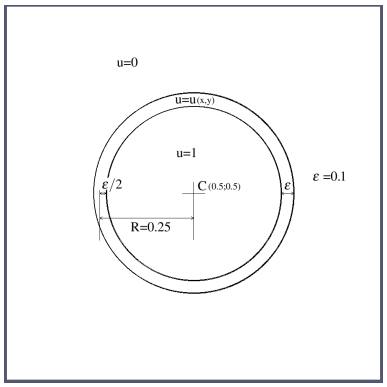

Figure 7: Circular-test-case-domain: sketch of the solution u.

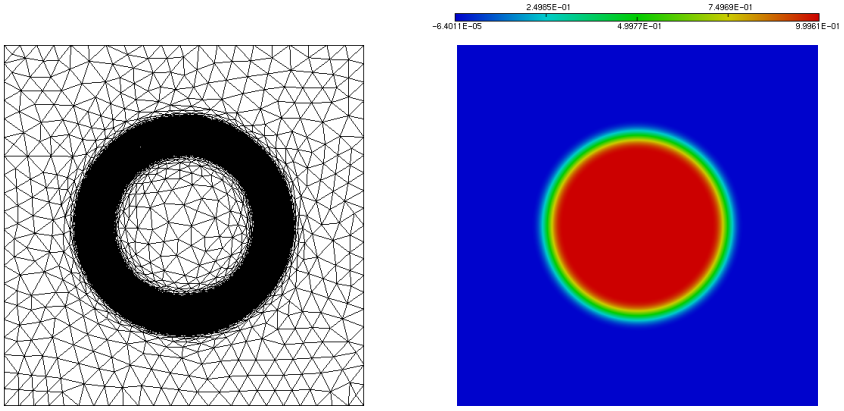

Figure 8: Circular layer test case: an adapted mesh and the corresponding numerical solution u_h

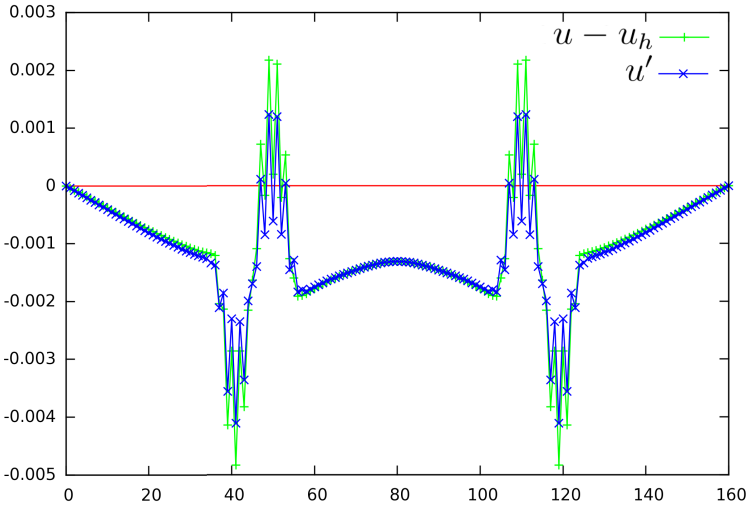

Figure 9: Thick bubble case: comparison of (+) error $u - u_h$ and (×) *a priori* corrector.

The three methods are again compared in Figure 11: standard FMG, Hessian-based adaptive FMG, norm-oriented adaptative. For the Hessian-based calculation, we observe a tendancy for a slower convergence for finer meshes, finishing with an error which is worst than the uniform refine-

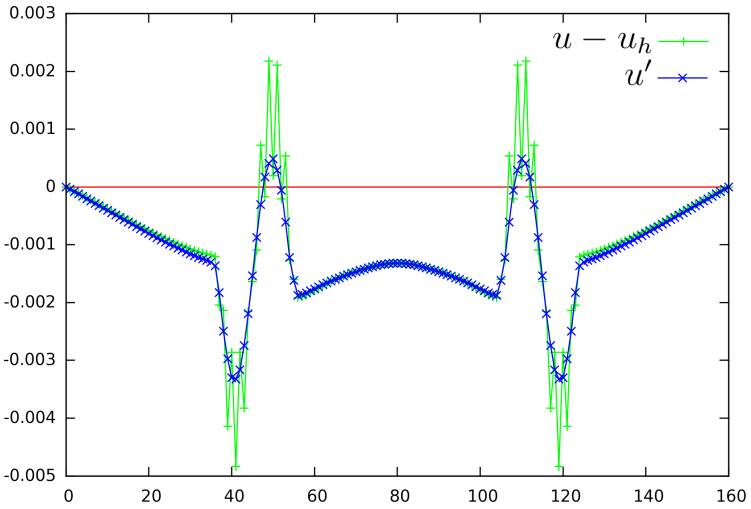

Figure 10: Thick bubble case: comparison of (+) error $u - u_h$ and (×) Defect-Correction corrector.

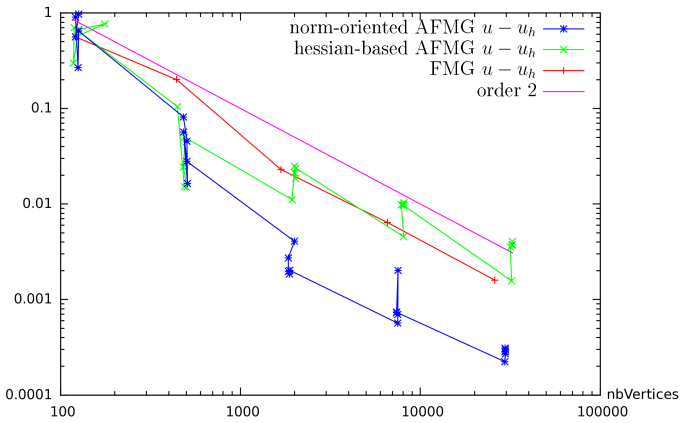

Figure 11: Thick bubble test case: convergence of the error norm $|u - u_h|_{L^2}$ as a function of number of vertices in the mesh for (+) non-adaptive FMG, (×) Hessian-based adaptive FMG, (∗) norm-oriented adaptive FMG.

ment. The proposed norm-oriented adaptive method behaves in a better way, with a five times smaller error than for the uniform refinement.

6.4. Bubble-like test case with thin interface

In order to evaluate the robustness of the methods with respect to steeper gradients, we consider the same test case, with a thinner transition: $\varepsilon = 0.02$. The convergence of the three methods is shown in Figure 13. Due to the very thin support of the right-hand side f, the three methods start with a zero f_h. Then, either with adaptation or refinement, the error increases to several units. The convergence of the uniform FMG shows an acceptable slope, but the error values are relatively huge. By comparison with the thick-bubble convergence, we may infer that the slope of uniform FMG will be second-order with even higher number of vertices and a 0.1 % error may be not attained for meshes of less than 10 millions nodes. The Hessian-based final result is a little better, but globally disappointing. The norm-oriented convergence starts chaoticly before being monotone and second-order for meshes finer than 2000 vertices. The final L^2 errors produced by the three methods are

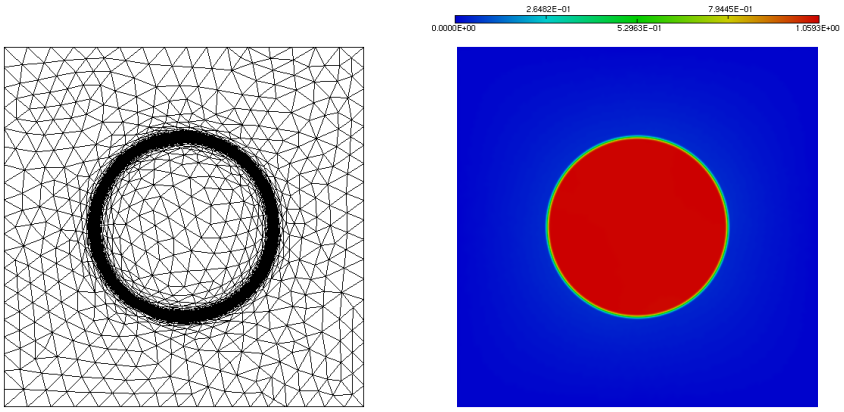

Figure 12: Thin bubble case: an adapted mesh and the corresponding numerical solution u_h

0.17589 with 25921 vertices for the uniform FMG, 0.03773 with 32127 vertices for the Hessian-based adaptation, and 0.000585 for 29742 vertices for the norm-oriented calculation, 300 times smaller than the first result.

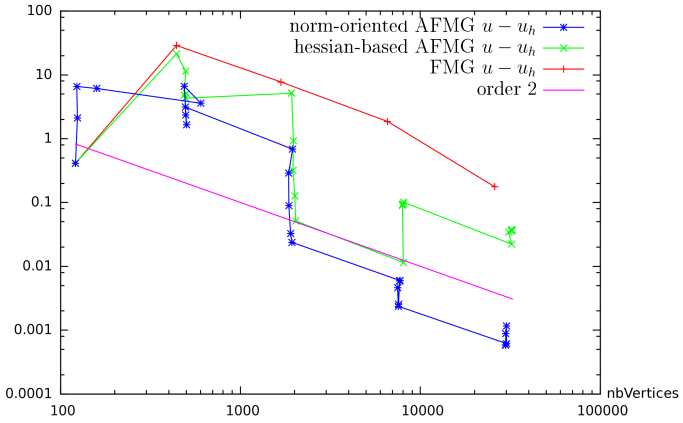

Figure 13: Thin bubble test case: convergence of the error norm $|u - u_h|_{L^2}$ as a function of number of vertices in the mesh for (+) non-adaptive FMG, (×) Hessian-based adaptive FMG, (∗) norm-oriented adaptive FMG.

6.5. Poisson problem with discontinuous coefficient

This test case addresses a non-regular case due to a discontinuous coefficient. The conditions are sketched in Figure 14a. The operator coefficient $1/\rho$ is 1000 times smaller in a vertical band of the computational domain, while the source term is 1000 times larger. The solution is nearly zero outside the central band and presents a smooth peak with maximum 1.26666 for $x = 0.5$, Figure 14b. Since no analytic solution is available, we take as reference solution the arithmetic mean:

$$u_{ref} = \frac{1}{2}(u^{Hess}_{100000} + u^{Norm}_{100000})$$

of the two adapted solutions (Hessian-based, norm-oriented) for 100000 vertices. The convergence is measured for the global integral $\int (u_h - u_{ref}) dxdy$.

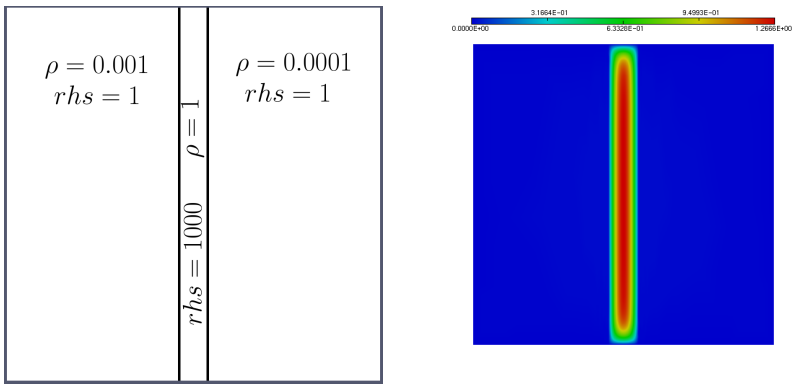

Figure 14: Discontinuity test case: (a) sketch of the inverse coefficient ρ and of the right-hand side rhs. (b) An accurate numerical solution.

Our computations are applied in the severe conditions of no *a priori* adaptation to the above discontinuities, as happens when the computation is internal to an identification loop. Then the comparison is again done with respect to a uniform mesh. The exact solution presents a discontinuous gradient and convergence on uniform meshes is only first-order. This strongly impacts the final error level of the uniform FMG which produces a convergence slope of order 1 and an error as large as 0.03 for 100000 vertices. The mesh-adaptive computations have also some trouble. They show a rather chaotic convergence. But, in the mean, we can say that convergence presents a slope around second-order. Errors for 30000 vertices can be estimated as around 10^{-4}. These results are particularly

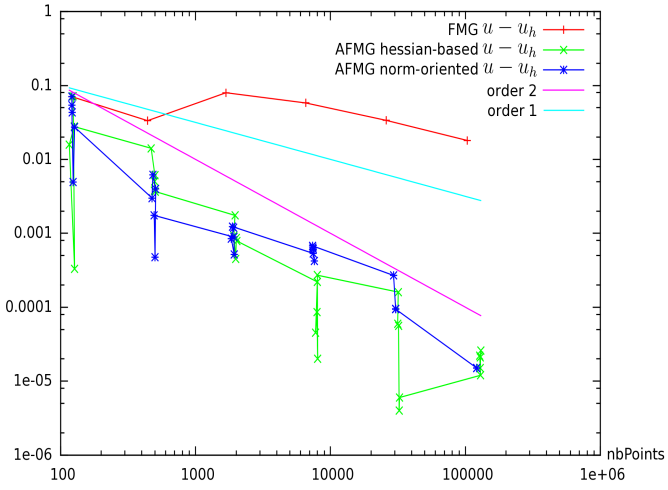

Figure 15: Discontinuity test case: convergences of $\int (u_h - u_{ref}) dxdy$ as a function of number of vertices in the mesh for (+) non-adaptive FMG, (×) Hessian-based adaptive FMG, (∗) norm-oriented adaptive FMG.

favourable to the two mesh-adaptation methods. The Hessian-based methods works very well and is also as already remarked, faster than the norm-oriented one which needs to solve three systems instead of one. The norm-oriented option is not CPU optimised. However, in the present case, it is interesting to see how the CPU cost behaves. In Figure 16, in 5000 seconds (workstation), the non-adaptative error is 0.03, the two mesh adaptative errors are of order 0.001.

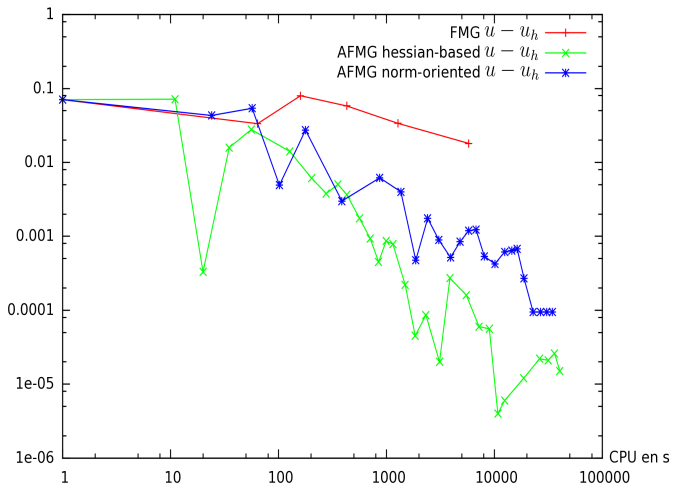

Figure 16: Discontinuity test case: convergences of $\int (u_h - u_{ref}) dxdy$ as a function of CPU time in seconds for (+) non-adaptive FMG, (×) Hessian-based adaptive FMG, (∗) norm-oriented adaptive FMG.

6.6. A 1D Boundary layer

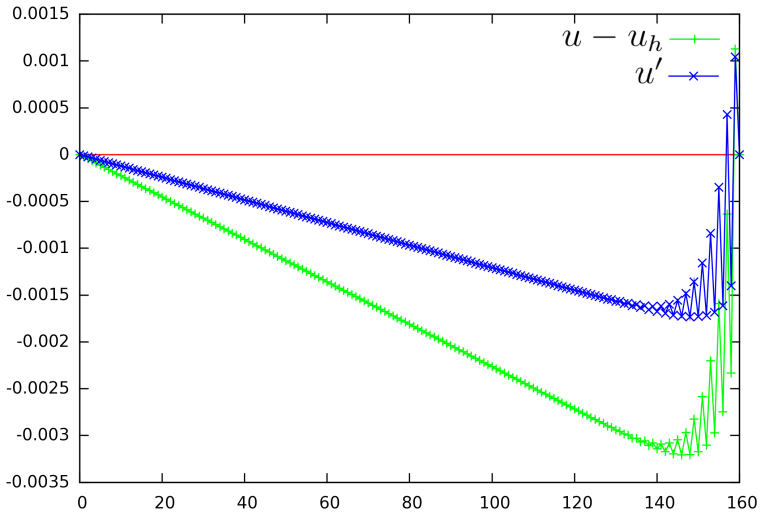

Figure 17: 1D Boundary Layer: comparison of error cuts for $y = 0.5$: plus signs (+) depict the approximation error $u - u_h$ and crosses (×) depict the *a priori* corrector u'_{prio}. The corrector is able to correct about 55% of the approximation error

The new method has shown a good behavior for all test cases we tried except one, which we describe now. It is a boundary layer case with a 1D solution: $u(x, y) = u(x)$ of a Dirichlet-Neumann problem $-\Delta u = rhs$ with

$$rhs(x, y) = (\alpha^2(exp(1/\alpha) - 1))^{-1} exp(x/\alpha) \quad ; \quad \alpha = 0.03.$$

We check first the correctors. Both seem adequate on a uniform grid, as shown with a horizontal cut depicted on Figure 17 and Figure 18.

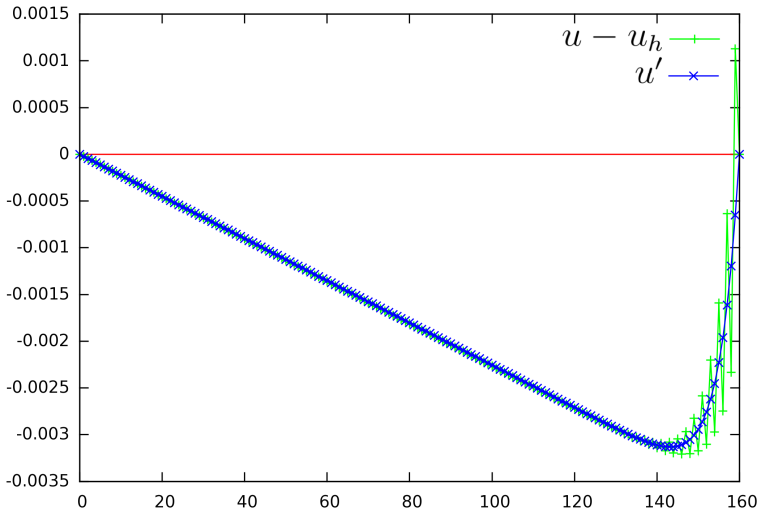

Figure 18: 1D Boundary Layer: comparison of error cuts for $y = 0.5$: plus signs (+) depict the approximation error $u - u_h$ and crosses ($\times$) depict the Defect-Correction corrector u'_{DC}. The corrector is able to correct about 80% of the approximation error.

In Figure 19, the non-adaptive FMG produces an approximation error of 0.003 (30000 vertices). This convergence is relatively satisfactory, being a second order convergence. However, in order to reach a 10^{-7} error level, several hundred millions vertices will be necessary with this sequence of uniform meshes. A second curve is obtained with the adaptative FMG with the Hessian-based criterion. Final convergence is disappointing since the slope is first-order. The same problem appears with our new algorithm. A deeper examination of adaptation criteria has shown that the high derivatives of the right-hand side f are very close to boundary $x = 1$. It could not be seen by the algorithm, because of the weighting by the adjoint u^*, which is zero at this boundary. We have replaced the norm-oriented optimum metric by its intersection with the metric based on the Hessian of f. Then the convergence improved a lot. In contrast, introducing the same metric intersection in the other cases did not produce second order convergence.

7. Conclusion

The norm-oriented mesh adaptation method is an answer to a well formulated problem : considering a numerical scheme, here the most used FEM, and prescribing an error norm, how to get the smaller approximation error in that norm, for a given number of vertices. The norm-oriented mesh adaptation method transforms the problem into an optimization problem which is mathematically well-posed. It relies on the following other features:

A corrector represents the approximation error. We give two examples of correctors. An *a priori* is built from the variational discrete statement. A Defect-Correction corrector is built from a finer-mesh

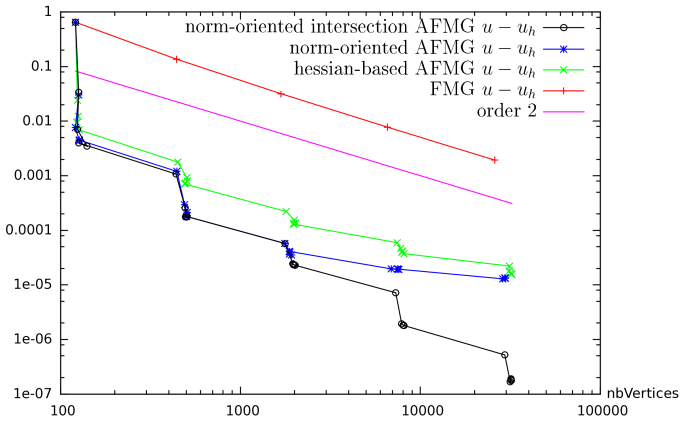

Figure 19: 1D Boundary Layer: convergence of the error norm $|u - u_h|_{L^2}$ as a function of number of vertices in the mesh for (+) non-adaptive FMG, (×) Hessian-based adaptive FMG, (∗) norm-oriented adaptive FMG, and (o) norm-oriented adaptive FMG with intersection with RHS Hessian.

defect correction principle. These correctors appear as not very accurate, but sufficiently accurate for our purpose. According to the type of approximation, at least the second one, Defect-Cooreection is extendable to many models and schemes.

The norm-oriented method is presented as a natural extension of the goal-oriented method, which, in our formulation, is itself a natural extension of the Hessian-based method. More precisely, while the Hessian-based method solves only the EDP under study, the goal-oriented method also solves an adjoint system (with linearised operator, transposed). The norm oriented solves three systems, a corrector (linearised system with an adhoc RHS), an adjoint (linearised and transposed, with the corrector as RHS), and the EDP itself. The three algorithms have in common an anisotropic *a priori* error analysis and a metric-based mesh parameterisation.

The Hessian-based method produces convergent solution fields but does not take into account the precise equation and discretization. The goal-oriented method takes into account equation and discretization, but is too focused on a particular output and does not produce convergent solution fields. The norm-oriented method has the advantages of both.

In order to show the improvement obtained with respect to previous methods, we compare in our experiments the two field-convergent options, Hessian-based and norm-oriented. Our benchmark examines convergence to continuous thanks to the application of a Full-Multigrid (FMG) process. Approximation errors can then be compared as functions of the number of degrees of freesom.

We have presented an example of comparison of computational effort. But our algorithm is not optimised. It does not enjoy the FMG effect neither for the corrector nor for the adjoint. It can be therefore much less efficient than Hessian-based adaptive FMG. We have indicated an idea to improve this issue. It is not of most interest to test this idea with a Poisson problem. The case of a compressible flow is being addressed now and will hopefully produce informations useful for practical computations in CFD.

For elliptic problems, the Hessian-based approach is nearly optimal as suggested by finite-element estimates. However the presented comparisons seem to indicate that the novel method carries a good improvement, especially for singular or stiff contexts. The method is rather general and we are now applying it to more complex PDE models from CFD (Euler, Navier-Stokes), involving dominant advection effects, for which a Hessian-based approach is much less efficient.

References

[1] A. Agouzal, K. Lipnikov, and Y. Vasilevskii. Adaptive generation of quasi-optimal tetrahedral meshes. *East-West Journal*, 7(4):223–244, 1999.

[2] F. Alauzet. *Adaptation de maillage anisotrope en trois dimensions. Application aux simulations instationnaires en Mécanique des Fluides*. PhD thesis, Université Montpellier II, Montpellier, France, 2003. (in French).

[3] E. Arian and M.D. Salas. Admitting the inadmissible: Adjoint formulation for incomplete cost functionals in aerodynamic optimization. *AIAA Journal*, 37(1):37–44, 1999.

[4] R. Becker and R. Rannacher. A feed-back approach to error control in finite element methods: basic analysis and examples. *East-West J. Numer. Math.*, 4:237–264, 1996.

[5] A. Belme. *Aérodynamique instationnaire et méthode adjointe*. PhD thesis, Université de Nice Sophia Antipolis, Sophia Antipolis, France, 2011. (in French).

[6] A. Belme, A. Dervieux, and F. Alauzet. A fully anisotropic goal-oriented mesh adaptation for unsteady flows. In *Proceedings of the 5th ECCOMAS CFD Conf.*, 2010.

[7] A. Belme, A. Dervieux, and F. Alauzet. Time accurate anisotropic goal-oriented mesh adaptation for unsteady flows. *J. Comp. Phys.*, 231(19):6323–6348, 2012.

[8] M. Berger. *A panoramic view of Riemannian geometry*. Springer Verlag, Berlin, 2003.

[9] G. Brèthes, O. Allain, and A. Dervieux. A mesh-adaptive metric-based Full-Multigrid for the Poisson problem. *submitted to I.J. Numer. Meth. Fluids*, 2014.

[10] M.J. Castro-Díaz, F. Hecht, B. Mohammadi, and O. Pironneau. Anisotropic unstructured mesh adaptation for flow simulations. *Int. J. Numer. Meth. Fluids*, 25:475–491, 1997.

[11] L. Chen, P. Sun, and J. Xu. Optimal anisotropic meshes for minimizing interpolation errors in L^p-norm. *Math. Comp.*, 76(257):179–204, 2007.

[12] J. Dompierre, M.G. Vallet, M. Fortin, Y. Bourgault, and W.G. Habashi. Anisotropic mesh adaptation: towards a solver and user independent CFD. In *AIAA 35th Aerospace Sciences Meeting and Exhibit*, AIAA-1997-0861, Reno, NV, USA, Jan 1997.

[13] A. Ern and M. Vohralík. Polynomial-degree-robust a posteriori estimates in a unfied setting for conforming, nonconforming, discontinuous Galerkin, and mixed discretizations. In *La Serena Numerica II, Octavo Encuentro de Anàlisis Numérico de Ecuaciones Diferenciales Parciales*, Enero 14 - 16, 2015. HAL Preprint 00921583, (2014).

[14] L. Formaggia and S. Perotto. Anisotropic a priori error estimates for elliptic problems. *Numer. Math.*, 94:67–92, 2003.

[15] M.B. Giles and N.A. Pierce. Improved lift and drag estimates using adjoint Euler equations. *AIAA paper*, 99-3293, 1999.

[16] R. Hartman. Multitarget error estimation and adaptivity in aerodynamic flow simulations. *SIAM Journal on Scientific Computing*, (31):708–731, 2008.

[17] W. Huang. Metric tensors for anisotropic mesh generation. *J. Comp. Phys.*, 204(2):633–665, 2005.

[18] A. Loseille. *Adaptation de maillage anisotrope 3D multi-échelles et ciblée à une fonctionnelle pour la mécanique des fluides. Application à la prédiction haute-fidélité du bang sonique.* PhD thesis, Université Pierre et Marie Curie, Paris VI, Paris, France, 2008. (in French).

[19] A. Loseille and F. Alauzet. Continuous mesh framework. Part I: well-posed continuous interpolation error. *SIAM Journal on Numerical Analysis*, 49(1):38–60, 2011.

[20] A. Loseille and F. Alauzet. Continuous mesh framework. Part II: validations and applications. *SIAM Journal on Numerical Analysis*, 49(1):61–86, 2011.

[21] A. Loseille, A. Dervieux, and F. Alauzet. Fully anisotropic goal-oriented mesh adaptation for 3D steady Euler equations. *J. Comp. Phys.*, 229:2866–2897, 2010.

[22] A. Loseille, A. Dervieux, P.J. Frey, and F. Alauzet. Achievement of global second-order mesh convergence for discontinuous flows with adapted unstructured meshes. In *37th AIAA Fluid Dynamics Conference and Exhibit*, AIAA-2007-4186, Miami, FL, USA, Jun 2007.

[23] F. Magoules. *Computational Fluid Dynamics.* CRC Press, Boca Raton, London, New York, Washington D.C., 2011.

[24] N.A. Pierce and M.B. Giles. Adjoint recovery of superconvergent functionals from PDE approximations. *SIAM Review*, 42(2):247–264, 2000.

[25] Y.V. Vasilevski and K.N. Lipnikov. An adaptive algorithm for quasioptimal mesh generation. *Comput. Math. Math. Phys.*, 39(9):1468 1486, 1999.

[26] Y.V. Vasilevski and K.N. Lipnikov. Error bounds for controllable adaptive algorithms based on a Hessian recovery. *Computational Mathematics and Mathematical Physics*, 45(8):1374–1384, 2005.

[27] D.A. Venditti and D.L. Darmofal. Adjoint error estimation and grid adaptation for functional outputs: application to quasi-one-dimensional flow. *J. Comp. Phys.*, 164(1):204–227, 2000.

[28] R. Verfürth. *A Posteriori Error Estimation Techniques for Finite Element Methods.* Oxford University Press, Oxford, 2013.

[29] M. Yano and D. Darmofal. An optimization framework for anisotropic simplex mesh adaptation: application to aerodynamics flows. *AIAA Paper*, 2012-0079, 2012.

[30] O.C. Zienkiewicz and J.Z. Zhu. The superconvergent patch recovery and a posteriori error estimates. Part 1: The recovery technique. *Int. J. Numer. Meth. Engng*, 33(7):1331–1364, 1992.